

Ein Modell zur simultanen Dimensionierung und Strukturierung von Fertigungssystemen mittels genetischer Algorithmen*

Jens Arnold¹, Thomas Fischer und Anja Fröhner²

¹ Technische Universität Chemnitz, Fakultät für Informatik, 09107 Chemnitz

² Technische Universität Chemnitz, Institut für Betriebswissenschaften und Fabrikssysteme, Erfenschlager Str. 73, 09125 Chemnitz

Zusammenfassung. Das Problem der Planung von Produktionssystemen besteht im Wesentlichen aus der Ermittlung des Ressourcenbedarfs (Dimensionierung: Art und Anzahl der benötigten Maschinen, Anlagen, Puffer, Werker, ...), der logischen Zusammenfassung dieser Ressourcen (Segmentierung, *engl.* cell grouping) sowie deren topologisch optimaler Anordnung innerhalb der Segmente (*engl.* facility layout design). Bisher wurden die Phasen „Dimensionierung“ und „Strukturierung“ separat in einem sequenziellen oder iterativen Optimierungsprozess betrachtet. Diese Teilung scheint in Anbetracht der Komplexität des zu Grunde liegenden Optimierungsproblems notwendig, ermöglicht allerdings nur das Auffinden suboptimaler Fertigungsstrukturen. Dieser Artikel stellt nach einer kurzen Darstellung bisheriger Fabrikplanungsmethoden einen ganzheitlichen Ansatz zur simultanen Dimensionierung und Strukturierung von Fertigungssystemen auf der Basis Genetischer Algorithmen vor. Dabei werden die bei herkömmlichen Verfahren einzeln untersuchten Partialprobleme als einheitliches Optimierungsproblem modelliert. Entscheidend für die Vergleichbarkeit unterschiedlicher Produktionsstrukturen ist die Quantifizierung ihres logistischen Aufwandes als Fitness-Wert. Deshalb wurde ein neuer Ansatz zur Nutzung logistischer Kenngrößen bei der Bewertung von Produktionsstrukturen untersucht, der weit über die bisher übliche Betrachtung des Transportaufwandes (Los/Meter) hinausgeht. Das Modell schließt eine verteilte simulationsbasierte Optimierung durch die parallele Bewertung der Produktionsstrukturen einer Generation ein. Ziel ist die Schaffung kostengünstigerer Fertigungsstrukturen für gegebene Produktionsprogramme unter Berücksichtigung der für die Planung verfügbaren Zeit. Am Beispiel der Strukturierung eines mittelständischen Unternehmens zur Fahrradproduktion (Beispieldaten der TU Linz) wird die Leistungsfähigkeit dieses Ansatzes gezeigt und ein Ausblick auf geplante Verbesserungen und Erweiterungen gegeben.

Schlüsselwörter. Genetischer Algorithmus, Layout-Design, Logistikkosten, Optimierung, Produktionssystemplanung

* gefördert durch das DFG-Innovationskolleg INK 17/A1-1 „Bildung eines vernetzten Logistik- und Simulationszentrums“

1 Problemstellung

Fertigungen lassen sich nur bis zu einer bestimmten Größe (Zahl der Fertigungsplätze, Zahl der Werker, Größe der Fläche) kostengünstig leiten und steuern. Wird diese Größe überschritten, ist es zweckmäßig, teilweise auch erforderlich, solche Fertigungen in kleinere Einheiten zu strukturieren bzw. zu segmentieren. Dies kann in vielfältiger Weise und in unterschiedlicher Tiefe produkt- bzw. objekt- und/oder verfahrensorientiert erfolgen. Konventionelle, tayloristische, technik- oder humanzentrierte Ansätze werden in diesem Artikel nicht betrachtet.

Die objektstrukturierte Fertigung ermöglicht die räumlich-organisatorische Zusammenfassung der zur vollständigen Fertigung von Produktgruppen, Produkten oder Produktteilen (Baugruppen, Unterbaugruppen, Teilegruppen, Teilefamilien, Teilearten) erforderlichen Einrichtungen und des erforderlichen Personals und damit eine hohe Autonomie solcher Struktureinheiten. Im Idealfall entstehen relativ autonome Substrukturen als Fraktale, Fabriken in der Fabrik, Fertigungssegmente, Fertigungsinseln usw. mit all ihren Vorteilen.

Unter den Bedingungen der Klein- und Mittelserienfertigung mit einer umfangreichen Produktpalette lassen sich die Einsatzchancen für eine objektorientierte Fertigung dadurch erhöhen, dass als Objekte Prozessteilegruppen gewählt werden, wobei die Gruppenrahmen nicht fest vergeben werden (wie zum Beispiel Zahnräder, Deckel usw.), sondern diese so bestimmt werden, dass eine Minimierung der durch die objektorientierte Strukturierung erforderlichen zusätzlichen Fertigungseinrichtungen sowie der Mehraufwendungen¹ für Transport und Steuerung durch die Heterogenität des Prozesses innerhalb und zwischen den Struktureinheiten erfolgt. Eine solche Teilegruppierung ist dann allerdings nicht mehr die Basis einer aufeinander folgenden Dimensionierung und Strukturierung der Fertigung, sondern das Ergebnis einer simultanen Dimensionierung und Strukturierung auf der Basis der einzelnen Teilearten, wie sie in konsequenter Auslegung des Stufengrundsatzes der Planung gefordert wird /FÖRS-83/.

Von den zurzeit bekannten Strukturierungsverfahren ermöglichen nur die kombinatorischen Verfahren auf der Basis von Teilearten die Verfolgung einer solchen Zielstellung (Bild 1). Selbst bei Einsatz von moderner Rechentechnik werden sie auch in absehbarer Zeit nicht für umfangreiche Teilesortimente einsetzbar sein, da die Anzahl der möglichen Lösungen exponentiell zur Anzahl der Fertigungsplätze steigt².

Einen viel versprechenden Lösungsansatz für eine objektorientierte Strukturierung unter Beachtung der praktischen Restriktionen bezüglich Rechenzeit und hinreichender Optimalität der Lösung stellen genetische Algorithmen dar (Bild 4), auf deren Basis das Softwaretool SiDiS^{GA}³ entwickelt wird.

¹ Mehraufwendungen entstehen durch Fluss entgegen der Hauptflussrichtung, die Nichtbelegung von Fertigungsplätzen durch einzelne Teilearten und durch Wechsel der Fertigungsaufträge zwischen Fertigungssystemen.

² Bei 10 Fertigungsplätzen existieren bereits über drei Millionen zulässiger Lösungen.

³ **S**imultane **D**imensionierung und **S**trukturierung mit **G**enetischen **A**lgorithmen

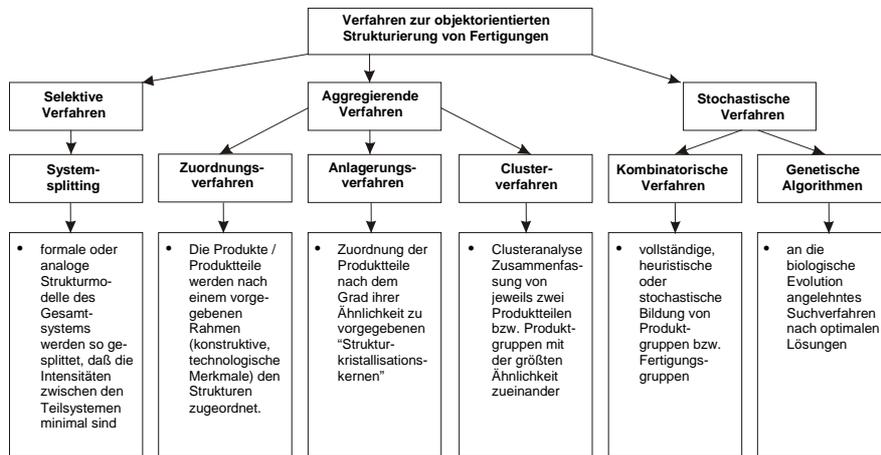


Bild 1: Verfahren zur Strukturierung von Fertigungssystemen

2 Methodische Grundlagen der genetischen Algorithmen

Die Optimierung von Systemen mit praxisrelevanten Dimensionen frei veränderbarer Parameter ist mit den klassischen Verfahren der Analytik und Kombinatorik nicht zufrieden stellend lösbar. Genetische Algorithmen ermöglichen es, die von SCHULTE und BECKER /SYDO-93/ erhobene Forderung, dass „Optimierungsalgorithmen (gewünscht sind), die in Abhängigkeit von einem vom Benutzer zu definierenden Ziel die Ergebnisse eines Simulationslaufes interpretieren und daraus automatisch Schlussfolgerungen für die Veränderung der Eingangsparameter des nächsten Simulationslaufes ableiten“ zu erfüllen. Dabei nutzen sie den bestehenden funktionellen Zusammenhang zwischen dem Zustand und den Parametern eines (natürlichen oder künstlichen) Systems und seiner Leistungsfähigkeit. Die stochastische Optimierung mittels Genetischer Algorithmen beruht auf dem biologischen Prinzip der Evolution: „Gute“ *Individuen* (Lösungen) überleben und geben ihre *Gene* (Variablen) an die nächste *Generation* (Lösungsmenge) durch Rekombination und Mutation weiter. Die theoretische Fundierung der Konvergenz eines solchen Iterationsprozesses wurde bereits in den Siebzigerjahren gezeigt /HOLL-75/ und in zahlreichen Anwendungen als praktikabel anerkannt /SCHA-89/, /FORR-93/, /ARNO-95/. Die heute verfügbare, leistungsfähige Rechentechnik macht die insbesondere für hohe Suchraumdimensionen mit großer Anzahl zu beachtender Nebenbedingungen prädestinierten Genetischen Algorithmen interessant zur Optimierung von Fertigungsstrukturen und -abläufen. „Aus der Sicht des Mathematikers, Informatikers und Ingenieurs stellt die Evolution ein extrem leistungsstarkes Optimierungsverfahren dar.“ /SCHÖ-94/.

Hier wird ein Genetischer Algorithmus für die Suche nach einer aufwandminimalen Fertigungsstruktur vorgestellt. Für diesen Zweck folgt nun die Formulierung der in Abschnitt 1 beschriebenen Problemstellung mit Begriffen der biologischen Evolution.

3 Der genetische Algorithmus

3.1 Codierung der Problemstellung

Ein Fertigungssystem (FS) besteht aus Fertigungsplätzen (FP), die bei Typgleichheit logisch zu Fertigungsplatzarten (FPA) zusammengefasst werden. Die Fertigungsaufgaben eines FP werden durch die Fertigungsvorgänge (FV) beschrieben, die in der zeitlichen Aufeinanderfolge der zu realisierenden Fertigungsaufträge die Fertigungsvorgangsfolge (FVF) bilden.

Eine Fertigungsstruktur wird mit dem Gruppierungs-, Struktur- und Fertigungsvorgangsfolgenchromosom (auch Parametervektor) beschrieben, die in diesem Tripel eine Lösung codieren. Dieser Chromosomensatz spannt den Suchraum S (auch Parameterraum) auf, in dem der Teilmenge P (*Population*) unter Beachtung der Nebenbedingungen (Lösungsrestriktionen) alle gültigen Lösungen I (*Individuen*) aus der Menge S (*Suchraum*) angehören.

$$P = \{I \in S : I \text{ ist eine gültige Lösung in } S\} \quad \text{(Gl. 1)}$$

Die Komponenten eines Chromosoms werden als *Gene* bezeichnet. Ihre möglichen Wertebelegungen sind die *Allele*. Bild 2 verdeutlicht diese Begriffe und Beziehungen.

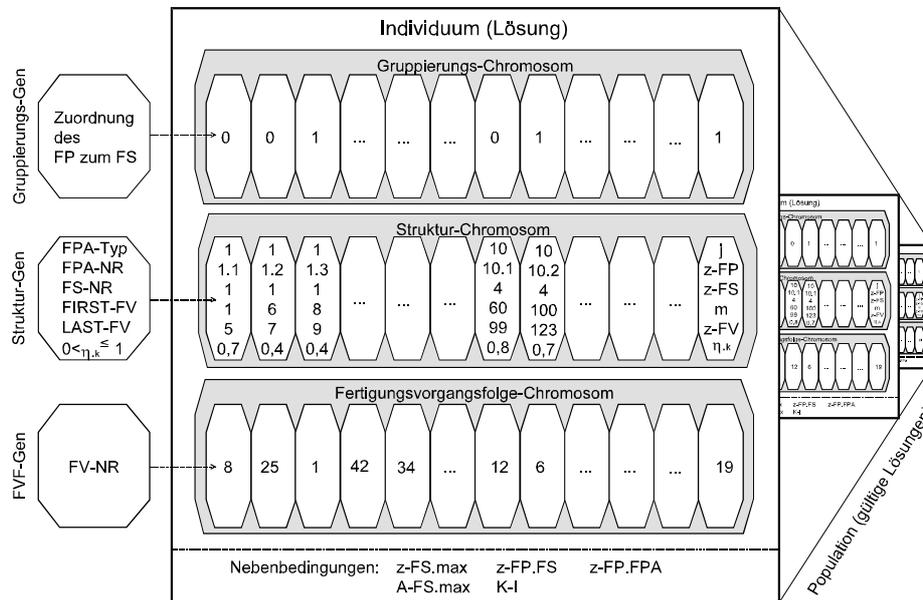


Bild 2: Codierung des Strukturierungsproblems

Legende:	
A-FS.max	maximale (verfügbare)Fläche je Fertigungssystem
FIRST-FV	Zeiger auf ersten Fertigungsvorgang eines Fertigungsplatzes
FP	Fertigungsplatz
FPA-NR	Nummer der Fertigungsplatzart
FPA-Typ	Typ der Fertigungsplatzart (Ordnungsbegriff)
FS	Fertigungssystem
FS-NR	Nummer des Fertigungssystems
FV-NR	Nummer des Fertigungsvorgangs
η_K	kapazitiver Nutzungsgrad
K-I	Investitionskosten
LAST-FV	Zeiger auf letzten Fertigungsvorgang eines Fertigungsplatzes
z-FP.FPA	Zahl der Fertigungsplätze der Fertigungsplatzart
z-FP.FS	Zahl der Fertigungsplätze des Fertigungssystems (maximal zulässig)
z-FS.max	Zahl der (maximal zu bildenden) Fertigungssysteme

3.1.1 Gruppierungschromosom

Die Gene dieses Chromosoms korrespondieren in ihrer Position mit denen des Strukturchromosoms und können nur die diskreten Allele 0 und 1 annehmen. Eine 0 bedeutet, dieser und der im Strukturchromosom unmittelbar nachfolgende Fertigungsplatz gehören zu einem Fertigungssystem. Eine 1 gibt an, dass dieser Fertigungsplatz der letzte Fertigungsplatz eines Fertigungssystems (in Hauptflussrichtung) ist und der im Strukturchromosom unmittelbar folgende Fertigungsplatz als Erster (in Hauptflussrichtung) in einem neuen Fertigungssystem steht.

3.1.2 Strukturchromosom (Reihenfolgechromosom)

In diesem Chromosom wird die topologische Anordnung der Fertigungsplätze codiert. Es repräsentiert also das Reihenfolgeproblem. Steht der Fertigungsplatz 1.1 vor dem Fertigungsplatz 1.2 im Chromosom, so entspricht diese Reihenfolge der Hauptflussrichtung innerhalb des Fertigungssystems.

Ein Struktur-Gen besteht aus dem FPA-Typ, der eindeutig den einzelnen Fertigungsplatz kennzeichnenden FPA-NR und der FS-NR, die eine eventuell vordefinierte Gruppierung einzelner Fertigungsplätze bestimmt. FIRST-FV und LAST-FV sind Verweise auf die diesem Fertigungsplatz zugewiesenen Fertigungsvorgänge im Fertigungsvorgangsfolgenchromosom. Der kapazitive Nutzungsgrad η_K ist eine reelle Zahl zwischen 0 (kein Fertigungsvorgang zugeordnet) und 1 (Fertigungsplatz ist voll ausgelastet).

Die Länge dieses Vektors (Anzahl der Gene) entspricht der durch die statische Dimensionierung bestimmten minimal zur Fertigung des gegebenen Produktionsprogrammes notwendigen Anzahl Fertigungsplätze plus einem definierten Mehrbedarf. Damit kann die Optimierung Fertigungsplätze hinzufügen, falls die damit verbundene Einsparung an Transportaufwand den Gesamtaufwand senkt.

3.1.3 Fertigungsvorgangsfolgechromosom

Die Länge dieses Chromosoms entspricht der Anzahl der Fertigungsvorgänge aller Fertigungsaufträge. Jeder Fertigungsvorgang wird durch eine eindeutige Nummer (Allel) bestimmt, mit der der entsprechende Fertigungsauftrag, die Stellung des Vorgangs im Arbeitsplan sowie die benötigte Kapazität verwaltet wird. Alle Gene mit den Fertigungsvorgängen eines Fertigungsplatzes stehen in Reihenfolge hintereinander. Das Gen des ersten und des letzten Fertigungsvorganges einer so zugeordneten Menge wird durch die im Strukturchromosom enthaltenen Verweise FIRST-FV und LAST-FV referenziert.

3.1.4 Weitere Parameter eines Individuums

Um bei der Erzeugung neuer Individuen das Einhalten der Nebenbedingungen

- Maximale Zahl der zu bildenden Fertigungssysteme (z-FS.max),
- Zahl der Fertigungsplätze je Fertigungssystem (z-FP.FS),
- Zahl der Fertigungsplätze je Fertigungsplatzart (z-FP.FPA),
- Maximale Fläche eines Fertigungssystems (A-FS.max) und
- Maximale Investitionskosten (K-I)

garantieren zu können, muss jedem Individuum noch ein entsprechender Vektor für die Speicherung dieser Werte zugeordnet werden.

Die Güte einer gefundenen Lösung wird durch die Fitness F (Zielfunktionswert) beschrieben⁴.

Generiert der evolutionäre Optimierungsprozess ungültige Lösungen, also Individuen, die nicht zur Population \mathbf{P} gehören, so erhalten diese Individuen einen überproportional hohen Aufwandswert (*Strafenfitness*). Auf die Berechnung der Fitnesswerte wird im Abschnitt 3.5 eingegangen.

3.1.5 Die Generation

Individuen, die gewissermaßen parallel untersucht werden und dabei direkt konkurrieren, bilden jeweils eine Generation. Die in der Literatur als sinnvoll angegebenen Werte für die Wahl der Generationsgröße (Anzahl der Individuen in einer Generation) haben sich in vielen Experimenten als Durchschnittswerte bewährt ($N=30$ /SCHA-89/, $N=50$ /SCHÖ-94/, $N>50$ /KINN-94/), sind aber für praxisrelevante Dimensionen der Fertigungsstrukturierung aus Gründen des Speicher- und Rechenzeitaufwandes nicht praktikabel. Mit der Beschränkung der Generationsgröße auf $N=21$ Individuen ist ein guter Kompromiss zwischen Konvergenzgeschwindigkeit der Fitnesswerte gegen das Optimum und dem Speicher- und Rechenzeitaufwand für Personalcomputer gefunden worden.

⁴ Fitness steht für Aufwendung, d.h. kleine Fitnesswerte sind anzustreben.

Die 21 Individuen einer Generation setzen sich aus 6 Eltern und den von ihnen erzeugten 15 Kindern zusammen. Das Verhältnis 6/15 von Eltern zu Kindern hat sich als besonders günstig für den Selektionsdruck erwiesen. Ein hoher Selektionsdruck vergrößert die Konvergenzgeschwindigkeit, senkt aber die Wahrscheinlichkeit, das globale Optimum zu finden und vergrößert die Gefahr, in einem lokalen Optimum „hängen zu bleiben“.

3.2 Schritte des evolutionären Optimierungsalgorithmus

In Bild 4 sind die einzelnen Schritte des Optimierungsalgorithmus dargestellt.

Im Schritt 1 wird eine Anfangslösung, die beliebig schlecht sein kann, an den Algorithmus in `LsgMenge[0]` übergeben. Diese Anfangslösung muss allerdings alle Nebenbedingungen (z.B. Anzahl der Fertigungsplätze pro Fertigungssystem), die vorgegeben sind, einhalten. Zur Beschleunigung der Konvergenz des Evolutionsprozesses kann eine (suboptimale) Anfangslösung nach einem Aggregationsverfahren gemäß Bild 1 vorbestimmt und an den genetischen Algorithmus übergeben werden.

Im Schritt 2 werden die weiteren Individuen der Elterngeneration `LsgMenge[1..5]` durch den genetischen Operator *Mutation* (siehe Abschnitt 3.3.3.) erzeugt.

Im Schritt 3 werden aus den 5 Eltern `LsgMenge[1..5]` plus dem bisher besten gefundenen Individuum `LsgMenge[0]` die Kinder generiert. Dabei wird für jedes Chromosom ein eigener genetischer Operator (*Reproduktion, Rekombination, Mutation*) angewandt. Die Wahrscheinlichkeiten, mit denen die jeweiligen Operatoren gewählt werden, sind im Programm festgelegt. Die Eltern werden deterministisch, gleichverteilt zur Fortpflanzung herangezogen.

Im Schritt 4 erfolgt mit der Aufwandsberechnung die Bestimmung der Fitnesswerte für alle Kinder (siehe Abschnitt 3).

Schritt 5 überprüft, ob eine Verringerung der Fitness eingetreten ist. Wenn dies der Fall ist, wird die bessere Lösung in `LsgMenge [0]` kopiert. In `LsgMenge[0]` steht immer die bisher beste gefundene Lösung mit dem geringsten Aufwandswert (beste Fitness).

Schritt 6 wählt aus den 20 Individuen der `LsgMenge[1..20]` 5 Eltern für die nächste Generation aus. Das geschieht nach dem *Roulette-Verfahren*. In diesem Verfahren werden nach dem Prinzip des Roulette-Rades die einzelnen Individuen (Lösungen) auf einem Kreis abgetragen. Da eine Minimierungsaufgabe vorliegt (Aufwand für den Transport und die Fertigungsplätze sollen minimal werden), wird der schlechteste Fitnesswert der Generation (höchster Aufwand) von den Fitnesswerten der anderen Individuen dieser Generation subtrahiert. Die „neuen“ Fitnesswerte der einzelnen Lösungen bestimmen die Sektorgrößen der Individuen auf dem Roulette-Rad. Die Wahrscheinlichkeit, dass eine zufällig geworfene „Kugel“ in einen „großen“ Sektor fällt, also ein „gutes“ Individuum überlebt, ist demzufolge höher, als dass ein „schlechtes“ Individuum seine Gene weitervererben kann.

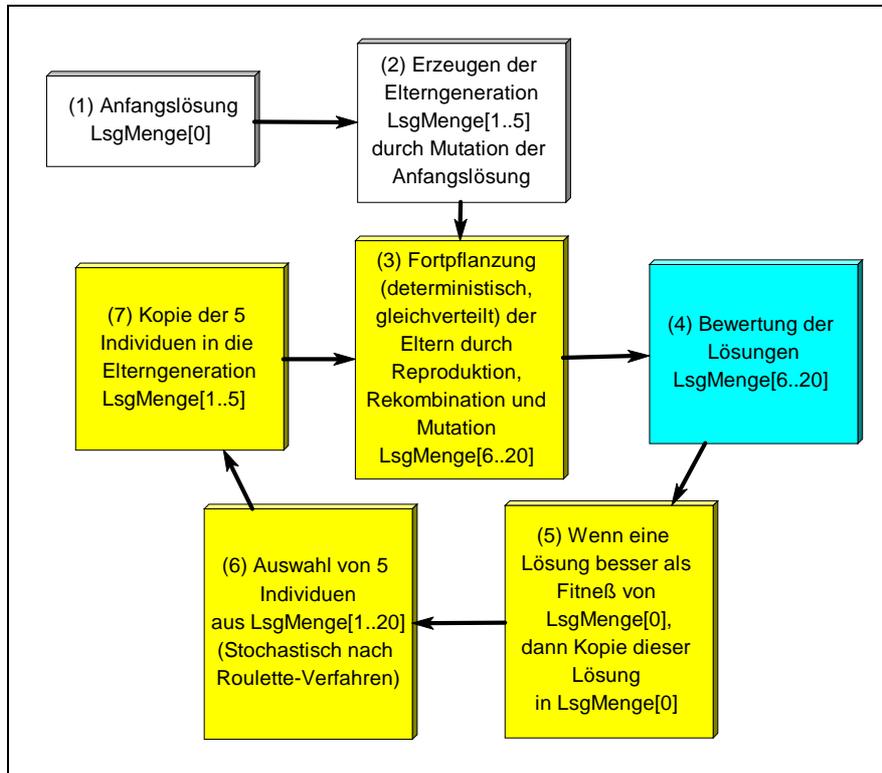


Bild 4: Der evolutionäre Optimierungsalgorithmus

Der Vorteil des Roulette-Verfahrens besteht darin, dass „gute“ Individuen bei der Vererbung bevorzugt werden, ohne allerdings „schlechtere“ Individuen völlig außer Acht zu lassen. Der Selektionsdruck (die mittlere Wahrscheinlichkeit, mit der ein Individuum in Abhängigkeit von seiner Fitness ausgewählt wird) dieses Verfahrens ist relativ gering. Ist die absolute Differenz der Fitnesswerte zwischen dem besten und dem schlechtesten Individuum einer Generation kleiner als $1/10$ der letzten Fitnessverbesserung, wird jedem Individuum, unabhängig von seiner tatsächlichen Fitness, eine Fitness zugeordnet, die direkt proportional zu seiner Rangordnung in der Generation ist. Dieses *Lineare Ranking*-Verfahren erhöht den Selektionsdruck und damit die Konvergenzgeschwindigkeit gegen das tatsächliche Optimum in seiner vermutlich bereits gefundenen, suboptimalen Umgebung.

In Schritt 7 werden nun die in Schritt 6 selektierten Individuen in die Elterngeneration $LsgMenge[1..5]$ kopiert.

Für die Wahl des Abbruchzeitpunktes liefert die Natur keinen Hinweis. Die biologische Evolution ist ein (möglicherweise unendlich) fortdauernder Prozess ohne erkennbares Endziel. Der Grund liegt im Sinn der Evolution: Anpassung der Lebewesen an sich permanent ändernde Umweltbedingungen. Da unsere zu optimierenden künstlichen Systeme aber ein konkret definiertes, funktionales Verhalten

in einer mehr oder weniger bekannten „Umwelt“ zeigen, sollte der Suchprozess nach dem Auffinden eines/des Optimums terminieren. Wie kann das erreicht werden?

- Der Abbruch erfolgt, wenn über eine bestimmte Anzahl Generationen keine Verbesserung des Fitnesswertes eingetreten ist.
- Der Abbruch erfolgt nach einer bestimmten Anzahl an Generationen. Die bis dahin errechnete beste Lösung wird als das Optimum angesehen.

Beide Varianten kommen im realisierten Programm in der Form zur Anwendung, dass der Nutzer jederzeit den Evolutionsprozess unterbrechen kann. Ansonsten beendet das Programm nach einer vorgebbaren maximalen Anzahl Generationen seit der letzten Fitnessverbesserung automatisch die Optimierung.

3.3 Die genetischen Operatoren

3.3.1 Reproduktion

Die Reproduktion ist die ständige Erneuerung und Wiederholung der genetischen Informationen. Die Chromosomen werden einfach von den Eltern übernommen. Die Reproduktion läuft deshalb bei allen Chromosomen gleich ab. Die Reproduktion garantiert die Stabilität der Lösungen, damit der Algorithmus nicht zu sehr im Suchraum „springt“.

3.3.2 Rekombination

Rekombinationen sind Prozesse, die zu einer Umgruppierung der genetischen Informationen führen. Durch die Rekombination werden genetische Informationen zwischen den Individuen (sexuelle Rekombination) und innerhalb der Chromosomen eines Individuums (asexuelle oder Selbstrekombination) ausgetauscht.

Beispielsweise werden zur Rekombination des Fertigungsvorgangschromosoms (Bild 5) zunächst zufällig, gleichverteilt zwei Fertigungsplätze des gleichen Typs

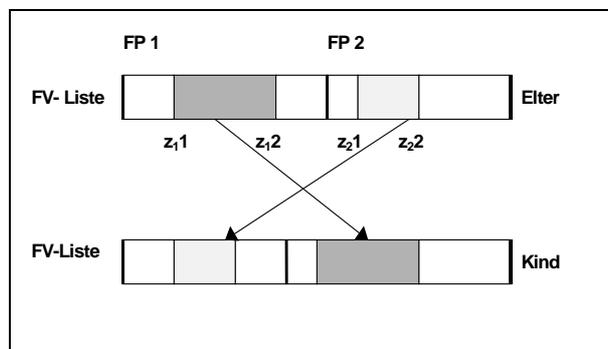


Bild 5: Selbstrekombination des Fertigungsvorgangschromosoms

ausgewählt. Aus den beiden Mengen der diesen Fertigungsplätzen zugeordneten Fertigungsvorgänge wird jeweils eine Teilmenge ausgewählt. Die Auswahl der Fertigungsvorgänge und deren Anzahl geschieht ebenfalls zufällig, gleichverteilt. In einem Iterationsprozess wird nun versucht, unter Hinzunahme und Wegnahme von Fertigungsvorgängen zu bzw. von den zufällig bestimmten Teilmengen, diese Teilmengen dem jeweils anderen Fertigungsplatz zuzuordnen.

Die Problematik dieses Operators besteht darin, dass nach dem Austausch der Teilmengen die Gesamtkapazität beider Fertigungsplätze nicht überschritten sein darf.

3.3.3 Mutation

Die Mutation ist der Zufall der Evolution und für die Anpassungsfähigkeit einer Population an sich ändernde Umweltbedingungen unverzichtbar. Die Mutation ist dafür verantwortlich, dass lokale Optima (suboptimal angepasste Individuen) wieder verlassen werden können und in andere Bereiche des Suchraumes „gesprungen“ wird.

Am Beispiel des Reihenfolgechromosoms wird eine Mutationsmöglichkeit erläutert. Zunächst wird der Fertigungsplatz z2 zufällig, gleichverteilt ausgewählt.

(a) Ist das Gen z2 nicht belegt, wird versucht, einen Fertigungsplatz hinzuzufügen. Dabei wird zufällig gleichverteilt eine Fertigungsplatzart ausgewählt. Zunächst wird geprüft, ob der ausgesuchte Fertigungsplatz noch hinzugefügt werden kann, d. h., die maximal mögliche Anzahl der Fertigungsplätze muss größer sein als die aktuelle Anzahl Fertigungsplätze dieser Fertigungsplatzart. Weiterhin darf die maximale Investitionssumme nicht überschritten werden. Zum Schluss muss der neue Fertigungsplatz noch ausgelastet werden. Dies geschieht auf folgende Weise (Bild 6):

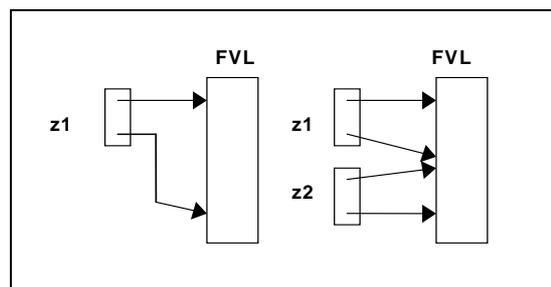


Bild 6 Aufteilung der Fertigungsvorgänge beim Einfügen eines Fertigungsplatzes

Zuerst wird wieder zufällig, gleichverteilt ein Fertigungsplatz z1 gleichen Typs ausgewählt. Ist dieser gefunden, wird kontrolliert, ob er mehr als 2 Fertigungsvorgänge besitzt. Wenn ja, werden seine Fertigungsvorgänge aufgeteilt und der neue Fertigungsplatz wird eingefügt.

(b) Ist das Gen z2 belegt, wird versucht, diesen Fertigungsplatz zu löschen. Ist der Fertigungsplatz einem Fertigungssystem fest zugeordnet, so darf er nicht gelöscht werden. Weiterhin muss geprüft werden, ob die minimal benötigte Anzahl Fertigungsplätze dieses Typs mit der Löschung nicht unterschritten wird.

Abschließend wird geprüft, ob sich die dem zu löschenden Fertigungsplatz zugeordneten Fertigungsvorgänge auf andere Fertigungsplätze aufteilen lassen. Ist eine Aufteilung der Fertigungsvorgänge möglich, kann der Fertigungsplatz gelöscht werden.

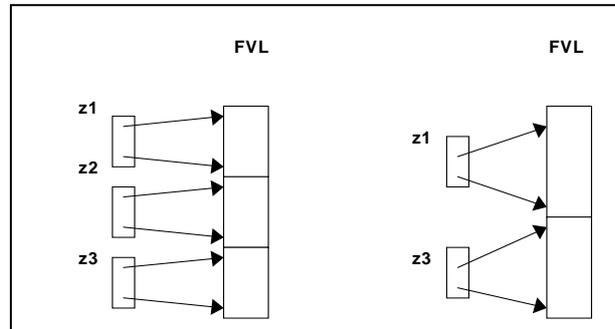


Bild 7: Aufteilung der Fertigungsvorgänge beim Löschen eines Fertigungsplatzes

3.4 Die Lösungsbewertung

Um zu entscheiden, ob eine von den Genetischen Algorithmen erzeugte Lösung gültig ist und ob sie als „gutes“ Individuum überlebt oder als „schlechtes“ Individuum abstirbt, wird diese Lösung in drei grundlegenden Schritten untersucht:

1. Ausschluss des Individuums wegen

- Überschreitung der zulässigen Investitionskosten,
- Überschreitung der zulässigen Zahl der Fertigungssysteme,
- Überschreitung der zulässigen Zahl der Fertigungsplätze je Fertigungsplatzart,
- Verletzung vordefinierter Gruppierung von Fertigungsplätzen,
- Zukauf nichtzulässiger Fertigungsplatzarten.

Dieser Schritt erfolgt im Genetischen Algorithmus und garantiert, dass an den Bewerter nur Individuen übergeben werden, die diese Restriktionen einhalten.

2. Verurteilung des Individuums wegen

- Überschreitung der zulässigen Zahl der Fertigungsplätze je Fertigungssystem,
- Überschreitung der zulässigen maximale Fläche eines Fertigungssystems.

Diese Individuen werden vom Bewerter mit einer Strafenfitness (extrem schlechter Fitnesswert) belegt und fallen somit als Elter aus.

3. Beurteilung des Individuums bezüglich

- Aufwendungen in Hauptflussrichtung,
- Aufwendungen entgegen Hauptflussrichtung,
- Aufwendungen bei Wechsel zwischen verschiedenen Fertigungssystemen,
- Aufwendungen durch Nichtbelegung von Fertigungsplätzen im Fertigungssystem,
- Aufwendungen durch Nichtbelegung in Reihenfolge der Fertigungsplätzen im Fertigungssystem.

Diese Individuen werden vom Bewerter mit einer Fitness belegt, deren Höhe über den Anteil am Roulette-Verfahren (Bild 4, Schritt 6) entscheidet und damit über die Wahrscheinlichkeit, mit der dieses Individuum als Elter der nächsten Generation fungiert.

Die Bewertung erfolgt an Hand des Zielfunktionswertes (Fitness) der konkreten Lösung (Individuum). Die Basis für die Berechnung des strukturbedingten Fitnesswertes ist die Anzahl der Transportlose.

Bezugsgröße für den Aufwand der im System enthaltenen Ausrüstungen bildet die Zahl der Fertigungsplätze. Die Berechnung des Fitnesswertes für die Fertigungsplätze ist notwendig, um einzuschätzen, ob durch den Zukauf eines Fertigungsplatzes tatsächlich eine Verbesserung der Struktur eingetreten ist, oder ob der Zukauf tatsächlich teurer ist als die Strukturverbesserung, die durch diesen Zukauf erzielt wurde.

Die Fitness F wird nach folgender Beziehung bestimmt:

$$F = F_{\text{Struktur}} + F_{\text{Fertigungsplätze}} \quad (\text{Gl. 2})$$

$$F_{\text{Struktur}} = F_{\text{HFR}} + F_{\text{HFR,e}} + F_{\text{FSW}} + F_{\text{NB,FS}} + F_{\text{NB,RF}} \quad (\text{Gl. 3})$$

Tabelle 1: Fitnessarten

Formelzeichen	Bedeutung	Inhalt
F	Fitness der Lösung	Zielfunktionswert der Lösung
F_{FSW}	Fitness bei Fertigungssystemwechseln	Aufwand, der durch Transport, Steuerung und Organisation je Flussbeziehung bei Wechsel zwischen des Systemen verursacht wird
F_{HFR}	Fitness in Hauptflussrichtung	Aufwand, der durch Transport, Steuerung und Organisation je Flussbeziehung in Hauptflussrichtung verursacht wird
$F_{\text{HFR,e}}$	Fitness entgegen Hauptflussrichtung	Aufwand, der durch Transport, Steuerung und Organisation je Flussbeziehung entgegen Hauptflussrichtung verursacht wird
$F_{\text{NB,FS}}$	Fitness durch Nichtbelegung von Fertigungsplätzen in	Aufwand, der durch Transport, Steuerung und Organisation je

Formelzeichen	Bedeutung	Inhalt
	Fertigungssystemen	Flussbeziehung durch Nichtbelegung von Fertigungsplätzen im Fertigungssystem je Fertigungsvorgangsfolge verursacht wird
$F_{NB,RF}$	Fitness durch Nichtbelegen von Fertigungsplätzen in der Reihenfolge der Fertigungsplätze des Fertigungssystems	Aufwand, der durch Transport, Steuerung und Organisation je Flussbeziehung durch Nichtbelegung von Fertigungsplätzen in Reihenfolge des Fertigungssystems je Fertigungsvorgangsfolge verursacht wird
$F_{\text{Fertigungsplätze}}$	Fitness für die Fertigungsplätze	Aufwand, der durch die einzelnen Fertigungsplätze verursacht wird (Lohn, Abschreibung usw.)
F_{Struktur}	Fitness für die Struktur	Aufwand, der durch die Struktur der Fertigungssysteme verursacht wird

4 Anwendungsbeispiel

Das Produktionsprogramm einer fiktiven Fahrradfabrik (Daten der TU Linz) besteht aus 3 Modellen (Mountain-, Trekking- und City-Bikes). Dafür müssen 65 Teilearten durch 433 Fertigungsvorgänge herstellbar sein. Bei der statischen Dimensionierung wurden 73 Fertigungsplätze (FP) in 46 unterschiedlichen Fertigungsplatzarten (FPA) ermittelt. Die Nebenbedingungen (Restriktionen) wurden wie folgt festgelegt

- maximale zusätzliche Investitionskosten 1 Mio. €
- maximal 10 Segmente
- maximal 50 FP / Segment
- maximal 10.000 m² Fläche / Segment

Die Optimierung mit SiDiS^{GA} führte zur Bildung von 5 Fertigungssegmenten (Bild 8). Die Anordnungsreihenfolge der Fertigungsplätze innerhalb der Segmente wird in Bild 9 exemplarisch am Segment „Felgenproduktion“ gezeigt.

Die Fitness der Ausgangslösung betrug 641,36 RGE⁵ (100 %). Die beste von SiDiS^{GA} ermittelte Lösung wurde mit 232,85 RGE (36 %) bewertet. Das tatsächliche Optimum beträgt 228,27 RGE (35 %), d.h. es wurde durch den Genetischen Algorithmus bis auf ca. 1% bezogen auf die Ausgangslösung der ungesplitteten Fertigung genähert!

⁵ Relative Geld-Einheiten

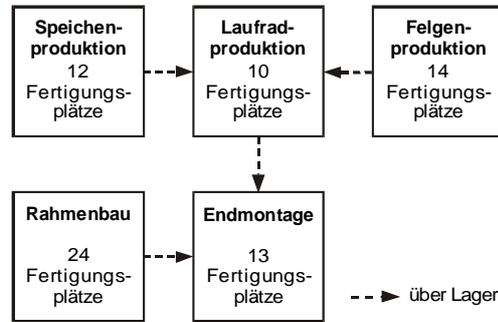


Bild 8: Optimal dimensionierte und strukturierte Fertigung als Ergebnis von SiDiS^{GA}

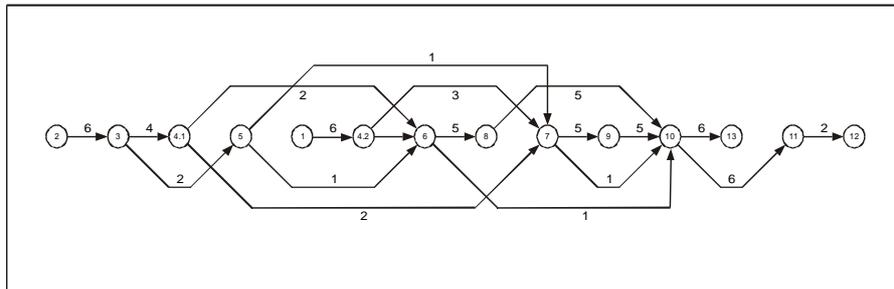


Bild 9: Beispiel-Segment „Felgenproduktion“. Keine Rücktransporte. Transporte über nicht belegte Fertigungsplätze optimiert auf 50 Lose.

5 Ergebnisse und Ausblick

Unsere Versuche haben gezeigt, dass mit den Genetischen Algorithmen bessere Strukturen erzielt werden als mit den herkömmlichen Verfahren. Dabei stellte sich heraus, dass es bezüglich der Rechenzeit günstig ist, die Individuen der ersten Generation (Urgeneration) mit Hilfe einer oder mehrerer „klassischer“ Verfahren (Bild 1) zu erzeugen.

Weitere Untersuchungen sollen Aufschluss über die Wirkung der Aufwandsfaktoren und deren Relationen, insbesondere auf das Strukturierungsziel bringen. Weiterhin wird die Stabilität der Lösung gegenüber der Änderung von Produktionsprogrammen und -sortimenten untersucht, um so zu Qualitätsmerkmalen der geplanten Struktur zu gelangen.

Mit einer verbesserten Feineinstellung der Ausführungswahrscheinlichkeiten für die genetischen Operatoren ist eine schnellere Konvergenz des evolutionären Optimierungsprozesses zu erwarten. Darüber hinaus kann, durch eine parallele Ausführung der Fitnessbestimmung für die einzelnen Individuen einer Generation, das Verfahren nahezu linear zur Anzahl verfügbarer Rechner/Prozessoren beschleunigt werden.

Literaturverzeichnis

- /ARNO-95/ Arnold, J.: Die Verwendung von Evolutionären Algorithmen bei der Optimierung von Fertigungssystemen.- Diplomarbeit TU Chemnitz-Zwickau, Fakultät für Informatik.- Chemnitz 1995
- /BANE-95/ Banerjee, P. und Zhou, Y.: Facilities layout design optimization with single loop material flow path configuration.- International Journal of Production Research, Vol. 33., No. 1, 1995, S. 183-203
- /EBLE-96/ Eblenkamp, M.: Planung von Produktionssystemen mit Evolutionsverfahren.- VDI-Fortschrittsberichte, Reihe 2, Nr. 396, Düsseldorf 1996
- /FAIF-96/ CAD-FAIF - Rechnergestütztes Fabrikplanungs- und Projektierungssystem. Dokumentation zur Einsatzcharakteristik. - TU Chemnitz, 1996
- /FORR-93/ Forrest, S. (Hrsg.): Proceedings of the fifth international conference on Genetic Algorithms.- Morgan Kaufmann Publishers, Inc., San Mateo 1993
- /FÖRS-83/ Förster, A.: Strukturierung von Teileflusssystemen der Fertigung im Maschinenbau.- Dissertation B, TH Karl-Marx-Stadt 1983
- /GEN-97/ Gen, M. und Chen, R.: Genetic Algorithms and engineering design.- Chapter 9: Facility layout design problems.- Wiley, New York 1997
- /HASS-94/ Hassan, M.M.D.: Machine layout problem in modern manufacturing facilities.- International Journal of Production Research, Vol. 32, No. 11, 1994, S. 2559-2584
- /HOLL-75/ Holland, J.H.:Adaptation in natural and artificial systems.-The University of Michigan Press, Ann Arbor 1975
- /KINN-94/ Kinnebrock, W.: Optimierung mit genetischen und selektiven Algorithmen.- R. Oldenbourg Verlag GmbH, München 1994
- /NEST-96/ Nestler, A.:Rechnergestützte Strukturierung von Fertigungssystemen auf der Basis ähnlicher Vorgangsfolgen mittels Genetischer Algorithmen.- Diplomarbeit, TU Chemnitz-Zwickau, IBF, Chemnitz 1996
- /RAJA-98/ Rajasekharan, M.; Peters, B.A.; Yang, T.: A genetic algorithm for facility layout design in flexible manufacturing systems.- International Journal of Production Research, Vol. 36, No. 1, 1998, S. 95-110
- /SCHA-89/ Schaffer, J.D. (Hrsg.): Proceedings of the 3rd International Conference of Genetic Algorithms & Applications.- Arlington 1989
- /SCHÖ-94/ Schöneburg, E., Heinzmann, F. und Feddersen, S: Genetische Algorithmen und Evolutionsstrategien.- Addison-Wesley, Bonn, Paris 1994
- /SYDO-93/ Sydow, A. (Hrsg.): Simulationstechnik - 8. Symposium in Berlin, September 1993.- Tagungsband, Friedr. Vieweg & Sohn Verlagsgesellschaft mbH, Braunschweig, Wiesbaden 1993